REMARKS

In view of the above amendments and the following remarks, reconsideration and further examination are requested.

By this amendment, claims 3 and 4 have been canceled, claims 9, 16, and 19 have been amended, and new claims 28-31 have been added. Therefore, claims 9-11, 16-19, and 24-31 are now pending.

Claims 3, 4, 9-11, 16-19, 21, 22, and 24-27 were rejected under 35 U.S.C. 102(e) as being anticipated by Lasserre (US 6,760,829). This rejection is traversed and is inapplicable to claims 9-11, 16-19, 21, 22, and 24-27 as amended for the following reasons.

Amended Claim 9 recites:

A data sharing apparatus comprising:

a data bus having a data width;

a memory;

a first-endian processor logically connected to said memory in a first-endian byte order via said data bus;

(A) a second-endian processor logically connected to said memory in a first-endian byte order via said data bus; and

(B) an address conversion unit operable;

(i) to invert values of two least significant bits of an address outputted from said second-endian processor and output an address including the inverted values to said memory when said second-endian processor performs a memory access for 8-bit data;

(ii) to invert a value of a second least significant bit of an address outputted from said second-endian processor and output an address including the inverted value to said memory when said second-endian processor performs a memory access for 16-bit data; and

(iii) to output an address from said second-endian processor to said memory without address conversion when said second-endian processor performs a memory access for data having the width of the first data bus,

wherein said memory stores structure data to be accessed by said first-endian processor and said second-endian processor,

said first-endian processor executes a first program that defines the structure data,

(C) said second-endian processor executes a second program that defines the structure data which includes data that is smaller than a basic word length, (C2) said data being defined in an order within the basic word length, (C3)and said order being in reverse to an order in the first program, and

(C1) said first-endian processor reads or writes the shared data to communicate with said second-endian processor and said second-endian processor reads or writes the shared data to communicate with said first-endian processor.

Independent claims 16, 19, and 28 also include similar limitations to those set forth above, and independent claims 29 and 31 include limitations similar to limitation C3 set forth above.

According to the above-mentioned claim limitation (A), the sharing of data having the width of the data bus between said processor and the processor of a different endianness, via the memory, can be implemented with an extremely simple structure (i.e., simply with a bus connection as recited in feature (A), rather than through a time consuming software conversion process).

According to the above-mentioned claim limitations (A) and (B), it becomes possible to access a memory which stores data of a different endianness, for data having a smaller width than the data bus. In addition, since the address conversion unit (as recited in feature B) can be configured through simple hardware, memory access speed is not reduced.

Through the combination of the above-mentioned claimed features (B) and (C1) to (C3), it becomes possible to share data with a smaller width than the data bus included in the above-mentioned structure data, between said processor and the processor of a different endianness. In addition, implementation with a simple structure, and without reducing access speed, becomes possible.

In summary, there is the effect of allowing data having the same width as the data bus as well as data having a smaller width than the data bus to be shared by a processor of different endianness, and in addition, preventing the occurrence of delay in the memory access of the processors.

The differences between Lasserre and the present invention are summarized in Table 1 and

Table 2 below.

TABLE 1

Memory access for data having the **same width** as the data bus

	Lasserre	Present invention	Particulars
Address conversion	No *0	No	*0 maybe
Data Positioning when where	Yes	Yes	*1 software routine (step 612 in Fig. 6; col. 10, lines 36 to38) *2 by (A) bus connection in a different endian byte order
	Endian Mismatch *1	Memory access	
	Software routine *1	Bus *2	
Structure Data in reverse order	No	No	
Memory access delay	Occurs *3	Does not Occur *4	*3 access with endian mismatch is aborted, and re-accessing is necessary. *4 accesses without delay

TABLE 2

Memory access for data having a smaller width than the data bus

	Lasserre	Present invention	Particulars
Address conversion	Yes *5	No *6	*5 by Memory control circuitry 708 (col. 10, lines 59 to 62) *6 by (B) the address conversion unit
Data positioning	Yes	Yes	*1 MMU/software routine (step 612 in Fig. 6; col. 10, lines 36 to 38) *7 by (A) bus connection in a different endian byte order *8 by (C2), (C3)
When	Endian mismatch *1	Memory Access *7	
Where	Software routine *1	Bus *7 & Structure data *8	
Structure data in reverse order	No	Yes *9	*9 by (C2), (C3)
Memory access delay	Occurs *3	Does not occur *4	*3 access with endian mismatch is aborted, and re-accessing is necessary. *4 accesses without delay

Tables 1 and 2 show the differences between Lasserre and the present invention, in terms of "address conversion", "data positioning", and "structure data in reverse order", and the "memory access delay".

"Address conversion" refers to a structure relating to the conversion of an address (least significant address bits) necessary for memory access by a processor of different endianness.

"Data positioning" refers to a structure relating to positioning of byte data necessary for memory access by a processor of different endianness.

"Structure data in reverse order" refers to a structure relating to the positioning of byte data necessary for memory access by a processor of different endianness.

"Memory access delay" is not a structure, but relates to one effect.

TABLE 1 shows the differences in the case where a processor of different endianness performs memory access for data having the same width as the data bus.

TABLE 2 shows the differences in the case where a processor of different endianness performs memory access for data having a smaller width as the data bus.

In TABLE 1, the difference in the structures of Lasserre and the present invention lies in the "data positioning". As a result, a difference in the effect produced appears in the "memory access delay".

Lasserre operates as follows when a processor of different endianness performs memory access for data having the same width as the data bus.

If there is an endianism mismatch (*1), then the request is aborted (*3) in step 610 and an abort handler is invoked. Based on the type of abort, the abort handler invokes a software routine (*1) that converts the data to an alternative endian format and rewrites the memory region in step 612. (FIG. 6; col. 10, lines 34 to 41).

In other words, when an endianism mismatch (*1) occurs, the initial memory access is aborted, and the endianism of the memory region is changed through the rewriting of the memory region. The processor needs to issue a memory access request again (*3).

In this manner, in Lasserre, memory access by a little-endian processor and a big-endian processor is made possible by changing the endianism of the memory region through a software routine.

However, since changing the endianism of the memory region through software takes time, there is a problem that, in the case where memory access is performed by a processor of different endianness, a significant delay occurs in the memory access.

On the other hand, in the present invention, with claimed feature (A), a second-endian processor is logically connected to the memory in a first-endian byte order via the data bus (in other words a bus connection in a different endian byte order than the endian byte order of the processor), an endianism mismatch is resolved merely by the passing of data through the bus.

Therefore, in the present invention, the memory access for data having the same width as the

data bus by a processor of a different endianness (the second-endian processor) is executed without the memory access being aborted, without address conversion by software, and without the occurrence of the aforementioned delay which occurs in the system of Lasserre by the claimed nature of the connection to the bus.

In TABLE 2, the differences in the structures of Lasserre and the present invention lie in the "address conversion", "data positioning", and "structure data in reverse order". As a result, a difference in the effect produced appears in the "Memory access delay".

The address conversion elements recited in amended Claims 9, 16, 19, and 28, have been amended to explicitly recite features (i), (ii), and (iii) set forth above. Such an address conversion unit (and method) is not disclosed in the Memory controller circuitry 706 which relates to "address conversion" in Lasserre (Col. 10, lines 59 to 62).

With regard to "data positioning", (i.e., the claimed features of the logical connection (A) to the bus and (C) defining of structure data in reverse order in claims 9, 16, 19, and 28), Lasserre performs the same process as in the case in TABLE 1. Since the judgment for endianism mismatch in FIG. 6 in Lasserre is not concerned with data width, the above process (FIG. 6; col. 10, lines 34 to 41) is carried out without distinguishing between memory access for data having the same width as the data bus and memory access for data having a smaller width than the data bus. In other words, the process details are the same for the case where the data width is the same as the bus and the case where data width is smaller. Therefore, as previously mentioned, in Lasserre, the aborting of the memory access, the changing of endianism through the rewriting of the memory region, and the re-accessing of the memory occur when a processor of different endianness performs memory access for data having a smaller width than the data bus.

Furthermore, although Lasserre suggests (col. 9, lines 30 to 33) solving data positioning by adding a data positioning function to the MMU and TLB which converts a logical address into a physical address, the specific structure thereof is not described. An implementation of this suggestion would require specialized hardware to position the byte positions of the data bus. In this case, delay caused by the positioning of the byte positions and the control thereof will occur.

On the other hand, the present invention implements "data positioning" (*7, *8) by claimed

features (A), (B), and (C3) recited in claims 9, 16, 19, 28, 29, and 31, i.e., (A) a second-endian processor logically connected to the memory in a first-endian byte order via said data bus (in other words, a bus connection in a different endian byte order), (B) said address conversion unit (or address conversion method), and (C3) the order being in reverse to an order in the first program.

For data defined by claimed feature (C3), data positioning can be implemented by merely passing the data through logical connection (A) to the bus.

Therefore, in the present invention, memory access for data having the same width as the data bus by a processor of different endianness (the second-endian processor) is executed without the aborting of the memory access and without the occurrence of the aforementioned delay.

In the present invention, "data positioning" is implemented through the combination of the logical connection (A) to the bus and (C3) defining of structure data in reverse order as recited in claims 9, 16, 19, 28, 29, and 31. This eliminates specialized processing for "data positioning" in the execution by the processor. Logical connection (A) to the bus ensures data pass-through, and (C3) defining of structure data in reverse order can be implemented by a programmer or a compiler. In other words, when two processors of different endianness are operating, the processors are freed from the processing for "data position".

In contrast, Lasserre merely suggests a structure which causes the processor to perform processing for "data positioning", in the operation of two processors of different endianness.

The present invention implements data sharing through a completely different approach and with a completely different structure, compared with Lasserre.

The Examiner argues that this point is disclosed in the structure data in FIG. 4 of Lasserre. However, Applicants believe that the Examiner's interpretation is incorrect because, if it were to be assumed that FIG. 4 in Lasserre disclosed structure data defined in reverse order, there would be a contradiction with the descriptions of the data positioning (col. 9, lines 30 to 33) and the memory region endian conversion (FIG. 6; col. 10, lines 34 to 41) in the Specification of Lasserre. In other words, when structure data that is defined in reverse order is applied in Lasserre, "data positioning" is applied twice over and memory access of data in the correct byte order will not be possible.

Therefore, Applicants assert that FIG. 4 in Lasserre illustrates register data, and not structure

data. In other words, FIG. 4 of Lasserre is properly interpreted as merely describing that in order to correctly share data, data should be shared in this manner in the register 404 of the big-endian processor 400, and the register 406 in the little-endian processor 402.

Furthermore, in contrast to Lasserre which discloses executing a specialized process (*1.) for data positioning during processor operation, the present invention does not require a specialized process for data positioning during processor operation.

The present invention solves the problem (memory access delay caused by specialized processing (*1)) which is not solved by Lasserre, and produces an effect that cannot be arrived at from Lasserre.

As discussed in detail above, Lasserre does not disclose each element of independent claims 9, 16, 19, 28, 29, and 31. Therefore, claims 9-11, 16-19, and 24-31 are not anticipated by Lasserre.

In view of the above, it is submitted that claims 9-11, 16-19, and 24-31 are allowable over the prior art of record and that the present application is in condition for allowance. The Examiner is invited to contact the undersigned by telephone to resolve any remaining issues.

Respectfully submitted,

Kazutoshi FUNAHASHI et al.

By: _____
Jeffrey R. Filipek
Registration No. 41,471
Attorney for Applicants

JRF/fs
Washington, D.C. 20006-1021
Telephone (202) 721-8200
Facsimile (202) 721-8250
January 28, 2008